Exploración De Datos Con Neo4j

Content

Juan Salvador Bahamón Arias
Data scientist / Omnibnk
Software Engineer / Universidad Distrital Francisco José de Caldas
jbahamon.arias@gmail.com
github/jusalbari/

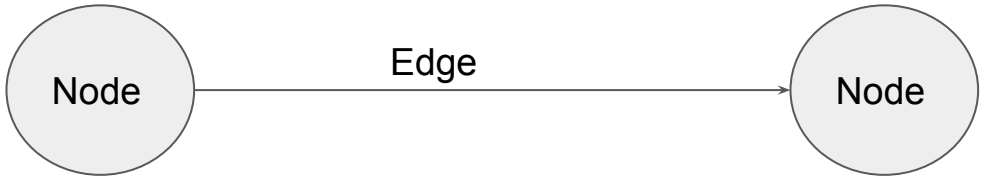
Some definitions

What is a database?

It’s a data collection

What is a graph?

It’s a structure that represent with points and lines the relationships between a couple of elements



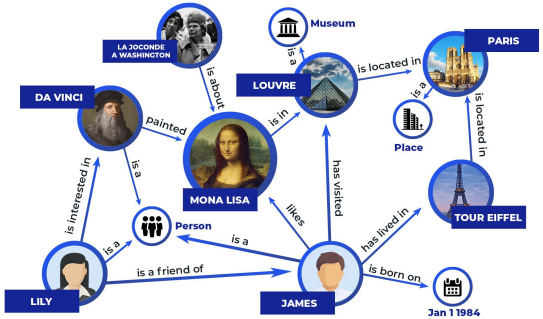
What is a graph database?

It's a collection of data that uses graph structures for semantic queries with nodes, edges, and properties to represent and store data.

Use cases with graph databases

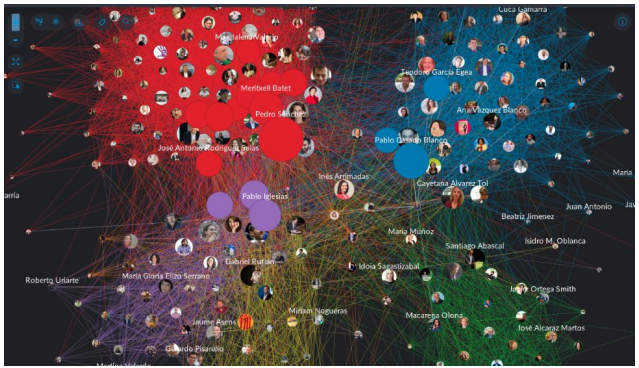
Fraud Detection

Knowledges Graphs



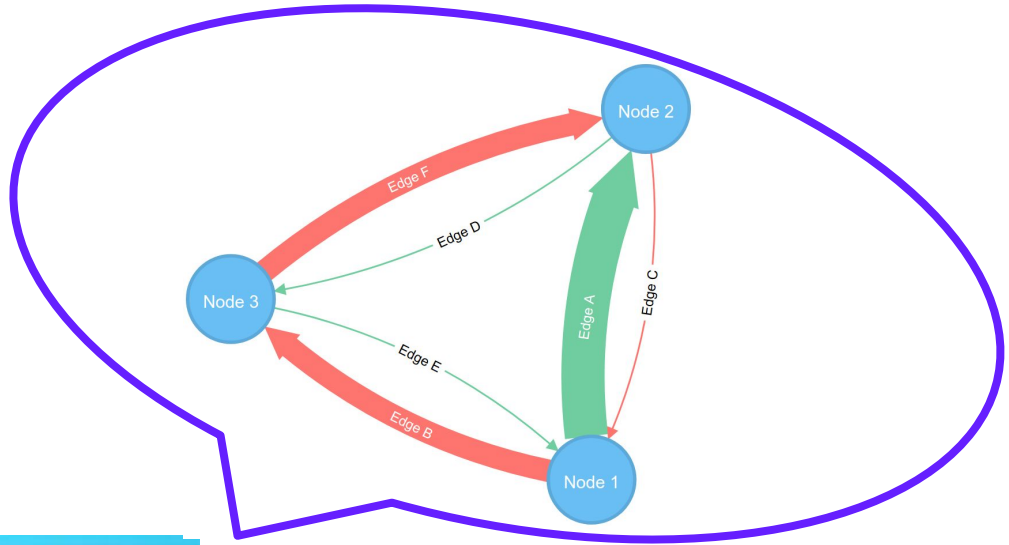
https://yashuseth.blog/2019/10/08/introduction-question-answering-knowledge-graphs-kgqa/

Social Networking

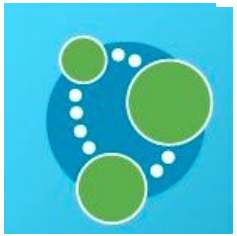


https://elpais.com/tecnologia/2019/11/14/actualidad/1573769942_710891.html

What is Neo4j?



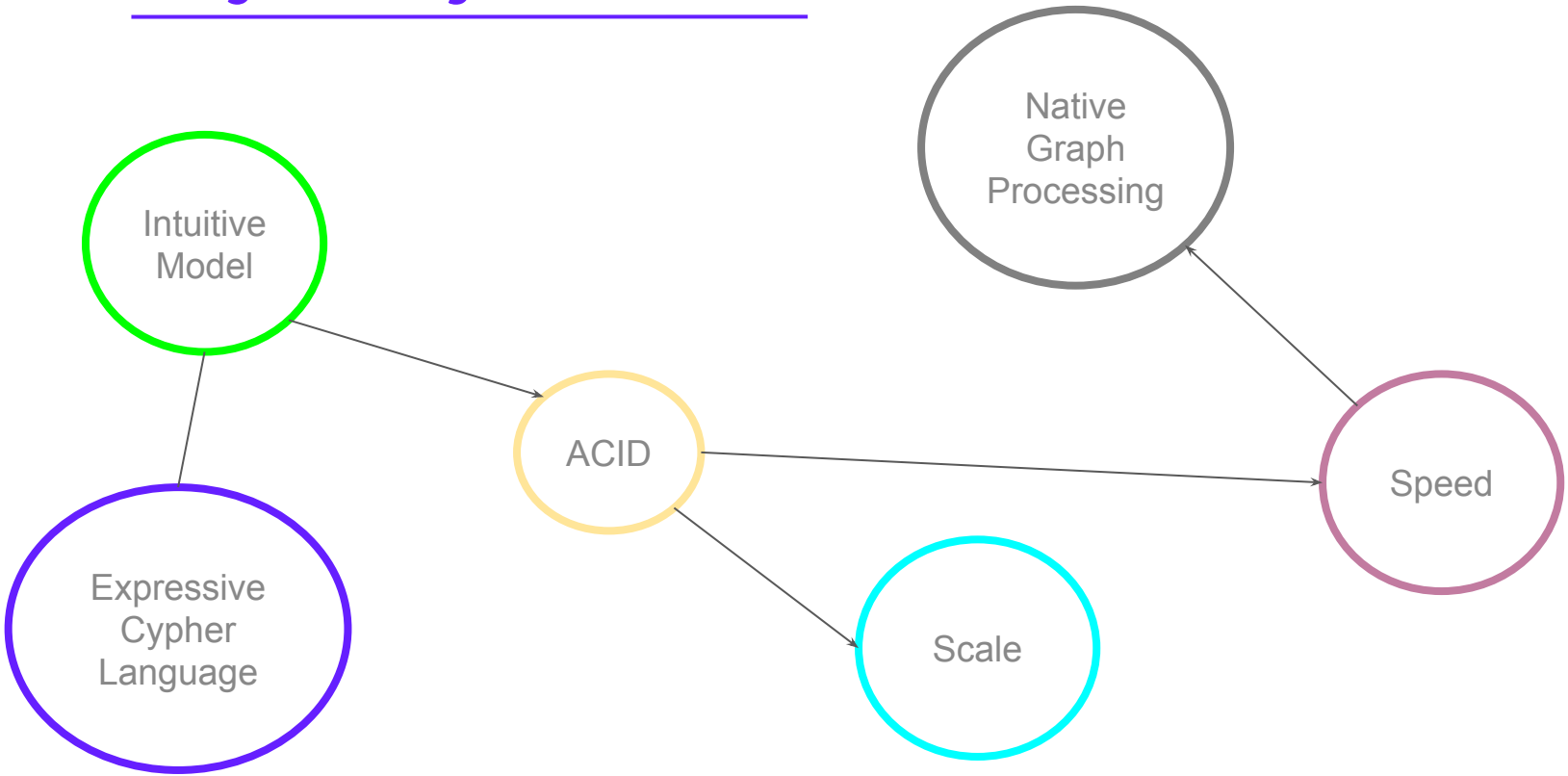
Since 2010
Language Java
Current version is 4.0.0
Open Source



“Unlike other databases, Neo4j connects data as it is stored, enabling it to traverse connections orders-of-magnitude faster”.

[https://neo4j.com/neo4j-graph-database]

Why Neo4j?

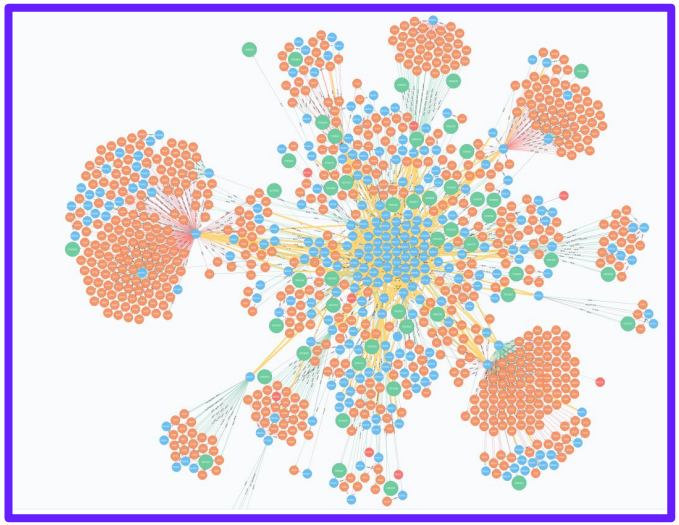

Native
Graph
Processing
Intuitive
Model
ACID
Speed
Expressive
Cypher
Language
Scale

Let’s Practice

Github

https://github.com/jusalbari/pycon2020

Let’s Practice



- Install Neo4j
- Install Python driver Py2neo
- Create database (nodes and edges)
- Queries structure
- Exploring data with Python

Install Ne4j

Install with Docker

```
sudo apt-get update

sudo apt-get remove docker docker-engine docker.io

sudo apt install docker.io

sudo systemctl start docker

sudo systemctl enable docker
```

Command

```
docker run --publish=7474:7474 --publish=7687:7687 --volume=$HOME/neo4j/data:/data --volume $HOME/neo4j/import:/var/lib/neo4j/import neo4j
```

Install on your pc

Opcion 1 - tar

https://neo4j.com/download-center/#community

```
tar -xf <filecode>.

cd neo4j-community-4.0.0/

bin/neo4j console
```

Opcion 2 - application

https://neo4j.com/docs/operations-manual/current/installation/

Install OpenJDK

https://mkyong.com/java/how-to-install-java-jdk-on-ubuntu-linux/

Default: **user**: neo4j // **password**: neo4j

Install py2neo, pandas and numpy

OMNI bnk

Opcion - pip

```
pip install py2neo
pip install pandas
pip install numpy
```

Opcion - conda

```
conda install -c conda-forge py2neo
conda install -c anaconda pandas
conda install -c anaconda numpy
```

Implementation in python

Import packages

```
import pandas as pd
from py2neo import Graph, Node, Relationship
```

Create Graph connection

```
graph = Graph(uri='bolt://localhost:7687',user='neo4j', password='neo4j')
tx = graph.begin()
```

Create nodes and edges (OGM)

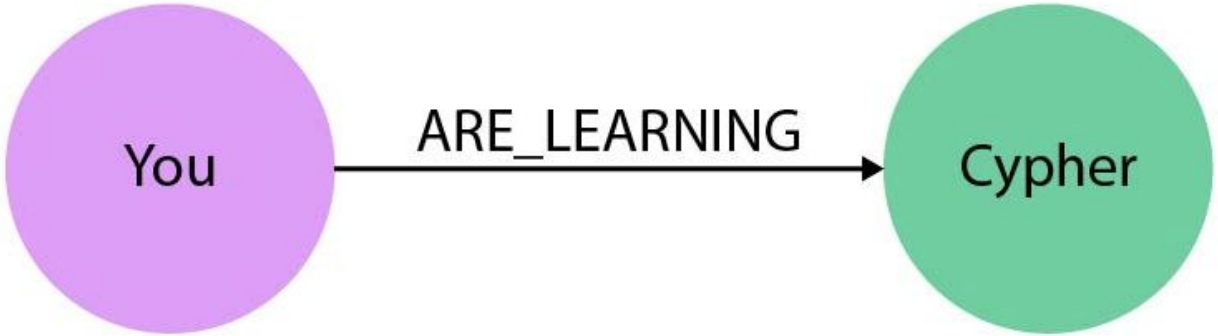
Create Nodes

```
node1 = Node("Node_type_1", name="Node 1", attr_2="value_attr2")
node2 = Node("Node_type_1", name="Node 2")
node3 = Node("Node_type_1", name="Node 3", attr_4=3 , attr_n=)
node4 = Node("Node_type_1", name="Node 4", attr4=23)
```

Create Edges

```
edge_a = Relationship(node1, "Edge A", node2)
edge_b = Relationship(node1, "Edge B", node3)
edge_c = Relationship(node2, "Edge C", node1)
edge_d = Relationship(node2, "Edge D", node3)
edge_e = Relationship(node3, "Edge E", node1)
edge_f = Relationship(node3, "Edge F", node2)
```

Cypher, the Graph Query Language



Match

```
Match (p:Person)-[:ARE_LEARNING]->(l:Language)
```

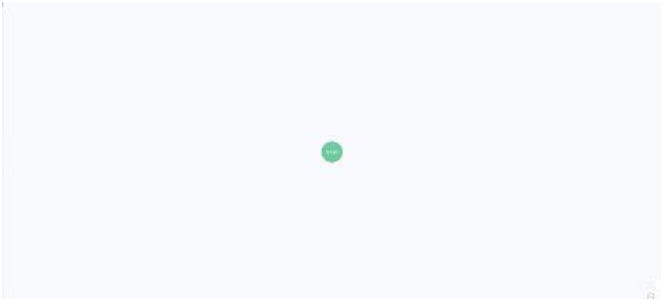
Match statement

Match option 1

```
Match (p:Person)-[:ARE_LEARNING]->(l:Language {name:'Cypher'} )
return p, l
```

Match option 2

```
Match (p:Person)-[:ARE_LEARNING]->(l:Language)
return p.name, l.name
```



Match option 3

```
Match (p:Person)-[:ARE_LEARNING]->(l:Language)
return p, l.name
```


OMNI
bnk
GRACIAS